Spatial-Temporal-Cost Combination based Taxi Driving Fraud Detection for Collaborative Internet of Vehicles

Xiangjie Kong, *Senior Member, IEEE,* Bing Zhu, Guojiang Shen, Tewabe Chekole Workneh, Zhanhao Ji, Yang Chen, and Zhi Liu

***Abstract*—Vehicle-to-vehicle (V2V) interaction and collaboration can provide us with a large number of mobile traffic trajectories that can be used to analyze driving behavior. In this paper, we propose a spatio-temporal cost combination based framework for taxi driving fraud detection (STC). First, the point of interest (POI) where taxis interact and collaborate with Collaborative Internet of Vehicles (C-IoVs) participants is identified, and a baseline trajectory model is built to determine the typical trajectory distribution. Second, a statistical model is used to calculate the travel distribution, travel time, and travel cost. At the same time, the taxi trajectory points are converted into evolving graphs to detect the abnormality of the local road segment. Then we can analyze the causes of outlier trajectories combined with the perception of abnormal road environments. Finally, the trajectories of real taxis were used to evaluate outliers, which proves the effectiveness and efficiency of the method.**

***Index Terms*—Collaborative Internet of Vehicles, Taxi driving outlier, Abnormal detection, Evolving graph**

I. Introduction

With the technology and application innovation emerge endlessly, Internet of Things (IoT) has become an important part of new infrastructure in many countries, which serves as a key infrastructural support in the development of the digital economy. As IoT moves toward smart-transportation, smart-industry, smart-health and other various industries, which makes cities smarter [1]. The Internet of Vehicles (IoV), an important branch of the IoT in the field of smart transportation, is a new industrial form of the transportation industry that deeply integrates electronics, information communication, transportation, artificial intelligence, etc., and is an effective method for urban vehicle trajectory supervision.

However, IoV has the problems of inconsistent and difficulty in integrating multi-category information, especially the large number of trajectories produced by taxis. Accurately detecting, identifying and correcting taxi drivers' outlier trajectories are of great significance to improving government traffic management [2], industry service quality, and passenger travel experience. Thence, Collaborative Internet of Vehicles (C-IoVs) is considered to be the solution to these problems. C-IoVs can connect roadside-units (RSU) deployed on traffic roads to achieve information exchange through wireless communication technology, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), etc., which can facilitate the modeling, monitoring, and optimizing of the whole process [3] in outlier driver trajectories accurately. Specifically, vehicles regularly share information with nearby sensing devices to obtain accurate neighborhood views, including nearby vehicle driving information, road traffic information, etc. [4]. Through the environmental perception of the signal light data, video surveillance data and radar data obtained by RSU in the C-IoVs, we can obtain information about some objective factors of the vehicle on the driving route, which provides a new opportunity to detect and classify taxi drivers' trajectories. The vehicles information interaction with the environmental perception in the C-IoVs, as shown in Fig. 1.

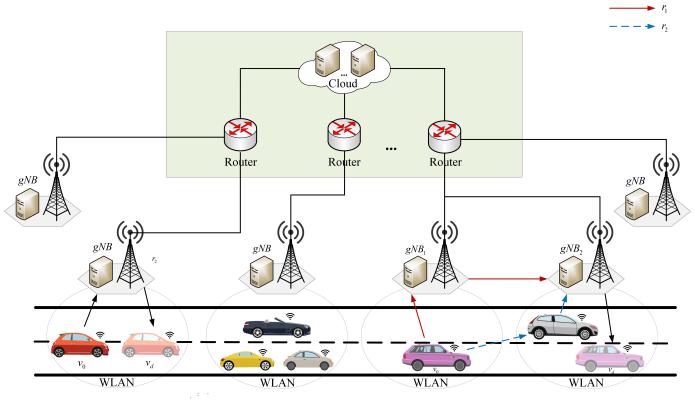


Fig. 1: Vehicle information interaction in C-IoVs.

Abundant trajectory data provides a deep foundation for abnormal trajectory detection. A route that is very different or inconsistent with the standard route is defined as an abnormal driving route [5]. In a static dataset, vehicle trajectory clustering [6] is a key technology for detecting abnormal behavior. Zhang et al. [7] adopt the mode-based iBAT algorithm to identify taxi driving outlier in modern cities. In addition, there may be a causal relationship between each trajectory point. Therefore, this type of relational dataset can be represented as a graph. Intelligent anomaly detection [8]

This work was partially supported by the National Natural Science Foundation of China (62072409, 62073295), Zhejiang Provincial Natural Science Foundation (LR21F020003), and Fundamental Research Funds for the Provincial Universities of Zhejiang (RF-B2020001).(Corresponding author: Guojiang Shen)

X. Kong, B. Zhu, G. Shen, Z. Ji, Y. Chen and Z. Liu are with College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: xjkong@ieee.org; BingZhu97@outlook.com; gjshen1975@zjut.edu.cn; jizhanhao.jzh@foxmail.com; YangChen@outlook.com; lzhi@zjut.edu.cn).

T. C. Workneh is with the Department of Computer Science, University of Verona, Verona, Italy (e-mail: tewabechekole.workneh@univr.it).

is crucial in evolving graph. Gupta et al. [9] introduce the concept of evolutionary community outlier in the dynamic subgraph. However, there are still two main shortcomings: it is generally believed that the long time or distance is an abnormal trajectory, without considering the actual vehicle operation; the other, most algorithms ignore the objective factors and lack of environmental perception of the actual city road network.

Based on the fact that V2R communication has high data storage and computational capacity, we focus on the study of real-time vehicles mobility, vehicle density, and abnormal behavior detection in large-scale urban environments. At the same time, the combination of V2V communication and V2R communication modes can improve the data transmission capacity and service quality, meanwhile solve the problems of large transmission delay in the communication link between vehicles. We propose a new algorithm for detecting abnormal trajectories and local road segments. First, we extract a valid trajectory and match it with the real city road network. Second, we extract position data from real-time. Then, the real-time data is used to convert into an evolving graph, where the detection of anomalies in the environment is perceived by the density change of the subgraph. Finally, the outlier trajectory is obtained based on the environmental perception. All in all, the contributions of this paper are as follows:

- We propose a method that combines the identified abnormal trajectory with environmental perception under C-IoVs to classify trajectories affected by objective geographic factors.
- We propose a conflict evidence fusion algorithm and consider the geographic constraints of the taxi trajectory and the collaboration between taxi trajectories in C-IoVs.
- We transform the static trajectory data set into a dynamic evolving graph network so as to perceive abnormal changes in the environment of the road segments. The algorithm is verified by a large number of taxis in the real world and prove the effectiveness.

In the following paper, we first present related work in Section 2. Then, in Section 3, we introduce some preliminary concepts and outline our algorithm, which is described in detail in Section 4. Section 5 presents our experiments and verifies the effectiveness of the proposed algorithm. Finally, in Section 6, we summarize our paper and provide an outlook for the future.

II. Related Work

We present a comprehensive survey of existing algorithms for detecting abnormal trajectories in urban traffic. We divide the existing approaches into two main categories. For spatial anomaly detection, similarity methods can be used to identify outliers based on distance or density, and cluster-based methods. For temporal anomaly detection, we consider the transformation of data into an evolving graph.

A. Spatial Anomaly Detection

The distance and density in the trajectory are updated in time intervals. As shown in Fig. 2, there are slight changes between adjacent trajectories, which means there are changes in distance and density between trajectories. Meanwhile, abnormal global/sub-trajectories can be detected in real-time.

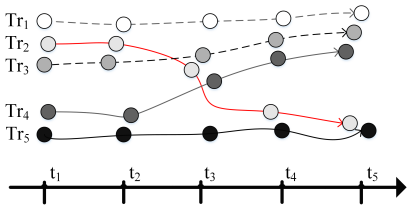


Fig. 2: Density and distance change of a series of trajectories with global/sub-trajectory detection.

Distance/Density Approaches In the detection process, an algorithm of distance measurement or field density calculation is used to find outliers. Ge et al. [10] establish a generative statistical model to describe distance distribution by combining travel evidence and travel distance evidence. Wang et al. [11] propose an abnormal trajectory detection and classification (ATDC) method. Although, these algorithms detect abnormal trajectories well by analyzing the distance and density distribution between trajectories, but didn't describe the maneuverability of driving behavior restricted by the actual road network.

Global/Sub-trajectory Processing Approaches Through an in-depth analysis of urban traffic applications, including flow, section flow, trajectory and its sub-trajectory to identify multiple types of anomalies, thereby significantly improving the effectiveness of anomaly detection et al. [12] , trajectory anomaly detection algorithms can be divided into two categories: global processing and sub-trajectory processing. For global processing, Zhu et al. [13] propose time-dependent popular routes based real-time trajectory outlier detection (TPRRO) method. Since passengers pay according to the meter, Zhou et al. [14] propose a new driving outlier recognition mode. This type of algorithm considers sufficient vehicle conditions, only detected the entire trajectory and didn't find the start and end positions of the abnormality.

Compared with global processing, sub-trajectory processing can identify abnormal sub-trajectory segments. Chen et al. [15] propose isolation-based to identify which sub-trajectories are anomalous. However, the previous method is difficult to detect the abnormal sub-trajectory of continuous multiple segments, therefore Yu et al. [16] propose a novel trajectory outlier detection algorithm based on common slices sub-sequence. Due to the time-varying, sparsity distribution of trajectory extraction and matching, it's still impossible to judge whether exists an objective reason in the abnormal trajectory detection.

B. Temporal Anomaly Detection

The evolving graphs have nodes, edges and attributes updated gradually over time, updates by time interval and continuous updates in a stream [17]. As Fig. 3. showed, the structure of graphs changes a bit between adjacent snapshots, which means a few of nodes and edges will be updated between snapshots.

Node-based Approaches One of the abnormal objects that can be found in a cooperation relationship between the edge nodes [18]. Compared with other nodes, irregular nodes are detected as abnormal nodes. Gupta et al. [9] introduce the concept of evolutionary community outlier (ECOutlier) to get

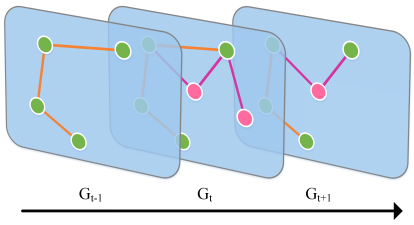


Fig. 3: The evolution of evolving graphs.

abnormal score of each node in the graph. The above algorithms all analyze the problem through individuals, ignoring that taxis are an inseparable group that influences each other.

Edge-based Approaches Compared with node-based anomalies, abnormal edges can be directly detected by the evolution of edge weights between nodes. Eswaran et al. [19] propose a SpotLight method based on random sketch to detect weighted in near real-time. Li et al. [20] achieves superior robustness against various types of noise and applys the scheme to the subspace clustering and graph recognition problems. Although a lot of work has been done in related papers, the relationship between upstream and downstream vehicles is ignored.

III. Overview

In this section, we will introduce some symbols, definitions, concepts, and an overview of the proposed algorithm. Our method is driven by typical actual deployment scenarios to build a "Perception-Fusion-Service" intelligent vehicle technology system. Starting from vehicle-level traffic management and control based on interactive analysis of intersection and vehicle information, outlier trajectory detection and cause classification are realized in data-driven scenarios.

A. Preliminary

Definition 3.1 (Irregular Outlier Trajectory). If the determined outlier trajectory does not occur in the environment of abnormal road segments, then it can be identified as an outlier caused by subjective factors of the driver, which we can determine as an irregular outlier.

Definition 3.2 (Driving Cost). With trajectory data, it is difficult to obtain the fuel consumption of the vehicle directly, so the various operations of the driver on the vehicle can indirectly indicate the fuel consumption. The driving cost is a three-dimensional array (spe, ang, dir) containing operations such as speed, angle, and direction

Definition 3.3 (Evolving Graph). Let $\mathcal{G}$ be a collection of road network crossroads during the trajectory time, which is the tuple $(\mathcal{S}, \mathcal{D}, \mathcal{T})$ and $\mathcal{S}, \mathcal{D}, \mathcal{T}$ respectively represent the set of origin points, destination points and time interval. For a trajectory Tr_n, exists $\mathcal{G}_t$ where $p_i = s_t \in \mathcal{S}_t, p_j = d_t \in \mathcal{D}_t$, s and d are the possibly time-evolving origin and destination nodes respectively in timestamp $t \in \mathcal{T}_t$.

Problem Definition For given taxi trajectory data, a pair of points of interest (POI) is selected and then a common trajectory is identified. If the trajectory (Tr) is sparse and different from the common trajectory, then it is considered an outlier. Similarly, if an interval subgraph $\mathcal{G}_t$ has a large density change, then we can consider the segment as an anomaly. Finally, the outlier trajectory and the anomalous road segment are merged to analyze in depth whether they are caused by human subjectivity or geographic objectivity.

B. Framework

We propose a framework that uses heterogeneous trajectory data to determine whether there is an outlier in driving behavior. The main structure of the framework is shown in Fig. 4. The three main parts are Trajectory Preprocessing, Spatial-Temporal-Cost Environment Combination and Classification. First, road network data is generated. Then, common trajectories are identified from a large number of trajectories in V2V/V2R communication, travel time, travel distance, and travel cost between trajectories are compared, then combine three aspects of evidence to obtain more reliable evidence for outlier detection. We calculate the density of each subgraph to develop changes in environmental perception. Finally, we combine the environmental perception to identify whether the abnormal trajectory is regular or irregular.

IV. Methodology

In this section, we explain the trajectory preprocessing, baseline modeling, a combined Spatial-Temporal-Cost method for taxi outlier detection and local segment anomaly detection, with the proposed framework shown in Fig. 4.

A. Trajectory preprocessing

Since the quality of driving trajectory data determines the accuracy of trajectory outlier assessment, it is very important to perform a series of data preprocessing when using large real-time trajectory data.

Trajectory Correction To visually represent the travel trajectory, it is necessary to use V2V communications to access the taxi's trajectory data with the assistance of other vehicles, while the actual position is retrieved from V2R to match the real road network. We calculate the speed and angle with real-time location interaction with V2V/V2R communication to correct the whole trajectory.

1) Detection Driving Speed The $d_{1,2}$ is the distance between two adjacent points in a trajectory, $v_{1,2}$ is the speed of two adjacent points. Among them, $p_1 = (lng_1, lat_1, t_1)$, $p_2 = (lng_2, lat_2, t_2)$, $p_i = (lng_i, lat_i, t_i)$ is physical location, which including longitude, latitude and timestamp, $\Delta\varphi$ is the mean value of the longitude of the two points, $\Delta\lambda$ is the mean value of the latitude of the two points and R is the long radius of the earth. The calculation method of distance and speed is as follows:

$$d_{1,2} = 2R \cdot atan2(\sqrt{sin^2(\Delta\varphi) + \Phi}, \sqrt{cos^2(\Delta\varphi) - \Phi}) \quad (1)$$

$$\Phi = (cos(lng_1 + lng_2) + sin(lng_1) \cdot sin(lng_2)) \cdot sin^2(\Delta\lambda) \quad (2)$$

$$v_{1,2} = d_{1,2}/(t_2 - t_1) \quad (3)$$

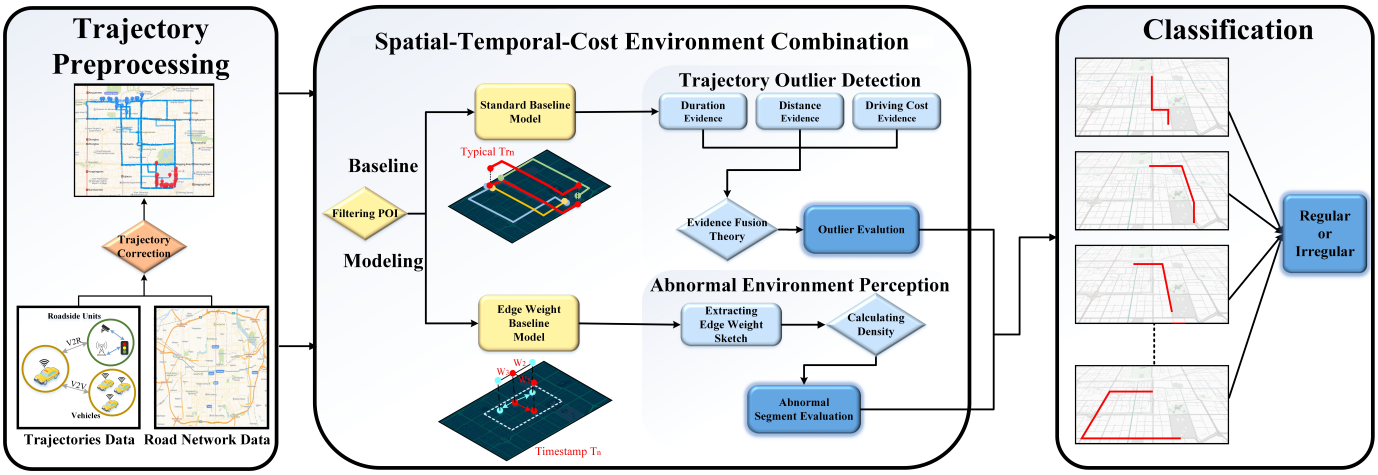


Fig. 4: Overview of the proposed framework.

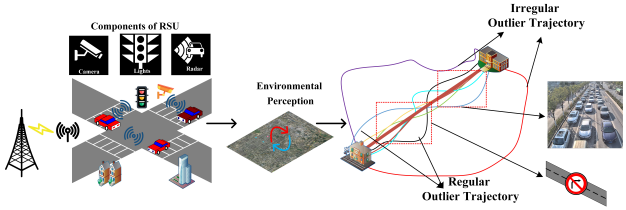


Fig. 5: Efficient vehicle-oriented management and control.

2) Detection Driving Angle $\theta_{1,2}$ is the clockwise included angle between two adjacent points in a trajectory.

$$\theta_{1,2} = atan2(sin(\Delta\lambda) \cdot cos(lng_2), \omega - \gamma \cdot cos(\Delta\lambda)) \quad (4)$$

$$\omega = sin(lng_2) \cdot cos(lng_1) \quad (5)$$

$$\gamma = sin(lng_1) \cdot cos(lng_2) \quad (6)$$

Trajectory correction is used to correct a group of directional latitude and longitude trajectories that may deviate from the road. The effect of trajectory correction is shown in Fig. 6. The red point is the original positioning point in V2V/V2R, while the blue is the positioning point after trajectory correction. (a) We successfully map the offset trajectory to the actual road network through correction. (b) Meanwhile, through interval sampling, the repeated, oscillating and offset points are eliminated and the trajectory is corrected to the actual road network.

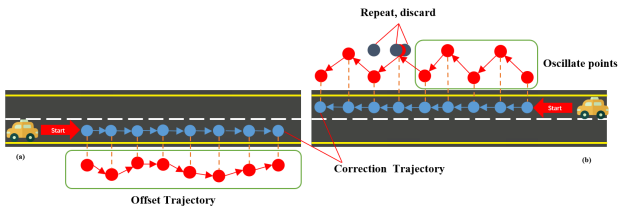


Fig. 6: Rendering of vehicle trajectory correction effect.

B. Spatial-Temporal-Cost Environment Combination

Baseline Modeling The baseline trajectory model is to capture the trajectory offset distribution of a pair of origin-destination nodes $< r_s, r_e >$ under normal driving. First of all, we must first screen the point of interest to obtain the origin-destination nodes $< r_s, r_e >$, then use two methods to obtain the baseline model.

1) Filtering POI The trajectory data used in this article is huge, complex, the origin and destination are chaotic. Therefore, it is necessary to manually select the frequently visited POI points to filter useless and sparse trajectories. The selected POI example is shown in Fig. 7.

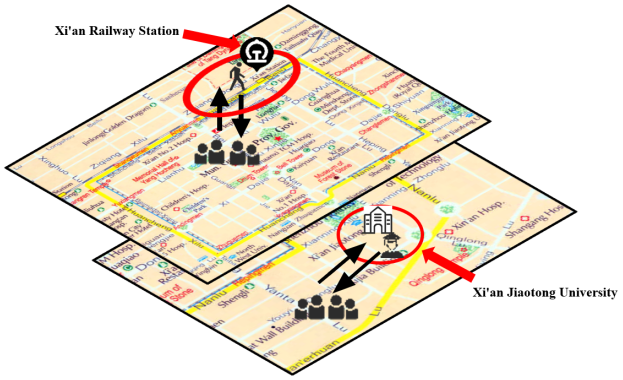


Fig. 7: The selected POI with environment interaction.

2) Standard Baseline Model The movement distribution of the aggregated trajectory [21] is used as a feature to evaluate whether there is an outlier event on the trajectory. The standard baseline model directly uses the shortest road shape in the pair of nodes $< r_s, r_e >$. We use Gaussian distribution to obtain M common trajectories under N trajectories between a pair of nodes under normal conditions. We assume that the trajectory distribution given a priori conditions of M commonly used typical trajectories $pr_i(i = 1, 2, \cdots, M)$ has the characteristics of Gaussian distribution parameters:

$$p(N|pr_i) = \mathcal{N}\left(N|\mu_i, \sigma_i^2\right) \quad (7)$$

where, $\mathcal{N}\left(N|\mu, \sigma^2\right)$ use the mean μ and variance σ^2 to represent the probability density function of the Gaussian distribution. For M pairs of parameters, we use the maximum likelihood estimation method to calculate.

3) Edge Weight Baseline Model Throughout the taxi trajectory, we take the passing intersection as the origin node in the subgraph and the next intersection as the destination node. The directed link of each subgraph is the actual direction of the trajectory and is given a relatively large weight, while the other two directions are given the same smaller weight. We can use a single coarse-grained feature, namely the total weight of the edge, to draw each graph.

Trajectory Outlier Detection In this module, we combine spatial-temporal-cost information with evidence theory fusion to detect outliers involving duration evidence, distance evidence, and travel cost evidence.

1) Duration Evidence A generative statistical model is used to model the travel time distribution of a pair of origin-destination nodes $< r_s, r_e >$, while the pseudo-code of duration evidence extraction for each trajectory is shown in Algorithm 1.

Algorithm 1: Driving Duration Evidence Extraction

Input: Trajectory Data Tr;
Output: Abnormal Duration Indicator t;
initialize $N = \varnothing$, $P = \varnothing$, $t = \varnothing$
for *each trajectory* tr_i *in Tr* **do**
  calculate the sum of absolute duration value n
  $N[i] = n$
end
find the index i_c of the common trajectory by N distrubution mannually
for *each duration* n_i *in N* **do**
  calculate probability p_i
end
for *each probability* p_i *in P* **do**
  calculete the abnormal indicator t
  $A[i] = t$
end

Specifically, we obtain the commonly used typical trajectories pr_{i} through the previous baseline model, assuming that a total of *N* different travel durations are identified in a pair of origin-destination nodes, while each duration is represented as n, with the observation result of travel time is the set of a priori conditions independent. Therefore, the observed value of travel time can be defined as:

$$p(N|PR) = \prod_{i=1}^{N} p(n_i|pr_i) \tag{8}$$

where $p(n_i|pr_i)$ is the conditional probability of the driving time observation value given the typical trajectory pr_i. *N* and *PR* are the collections of all travel time observations and common trajectory time respectively. After estimating the parameters, we define the abnormality of the trajectory with a travel duration of *t* according to the travel time as:

$$Abnormal(t) = 1 - \sum_{i=1}^{M} p(t|pr_i)p(pr_i) \tag{9}$$

2) Distance Evidence Statistical methods are used to calculate the probability distribution of the driving distance of the origin-destination nodes $< r_s, r_e >$. It is difficult to directly calculate the probability of driving distance distribution, therefore we use the full probability formula to divide the probability distribution. Specifically, we obtain common typical trajectories pr_{i} through the previous baseline model, assuming that a total of *K* types are identified in a pair of origin-destination nodes, which represent different travel distances, and each distance is expressed as *k*, the observation result of the travel distance is independent for the given prior conditions. Therefore, the observed value of travel distance can be defined as:

$$p(K|PR) = \sum_{i=1}^{K} \log\left(p(k_i|pr_i)\right) \tag{10}$$

where, $p(k_i|pr_i)$ is the conditional probability of the travel distance observation value given the common typical trajectories pr_i. *K* and *PR* are the collections of all travel distance observations and typical trajectories distances respectively. After estimating the parameters, in order to avoid the situation where the existence probability is 0, the logarithm based on *e* is taken in this paper. For a given prior probability, the abnormality of the trajectory with a travel distance *d* is defined as:

$$Abnormal(d) = \frac{1}{\prod_{j=1}^{M} e^{p(d|pr_j)p(pr_j)}} \tag{11}$$

3) Driving Cost Evidence Every operation of the driver changes the driving condition of the taxi and causes a change in the driving cost. The characteristic information such as location, speed, direction and steering angle should be fully considered when analysing the operating cost of the taxi. In this paper, the pseudo code of driving cost evidence extraction is shown for each trajectory in Algorithm 2.

As a consequence, a weighted multi-feature cost (*WMFC*) calculation method is proposed, as shown in the following formula:

$$WMFC(pr_i, Tr_j) = [speDis, angDis, dirDis] \times [w_1, w_2, w_3]^T \tag{12}$$

where $w_1 + w_2 + w_3 = 1$, pr_i is typical trajectories, Tr_j is the trajectory to be detected.

a) Speed Distance

The distance in speed $speDis(pr_j, Tr_i)$ refers to the difference between the average speed of the detected trajectory and the typical trajectory. By calculating the average speed difference of the two trajectory segments, the difference degree of the overall motion speed characteristics of the two trajectories is obtained. The calculation method is:

$$speDis(pr_i, Tr_j) = |\frac{1}{N_{pr_i}} \sum_{k}^{N_{pr_i}} v_{k,k+1} - \frac{1}{N_{Tr_j}} \sum_{k}^{N_{Tr_j}} v_{k,k+1}| \tag{13}$$

where $v_{k,k+1}$ indicates the speed between two adjacent points, N_{pr_i} represents the total number of speeds between two points on a typical trajectory, the same, N_{Tr_j} indicates the total number of speeds between two points on the detected trajectory.

b) Angular Distance

Algorithm 2: Driving Cost Evidence Extraction

Input: Trajectory Data Tr and Common Trajectory Index i_c;
Output: Driving Cost C;
initialize $Spe_0 = \varnothing$, $Ang_0 = \varnothing$, $Dir_0 = \varnothing$
initialize $Spe = \varnothing$, $Ang = \varnothing$, $Dir = \varnothing$, $C = \varnothing$
for *each trajectory* tr_i *in Tr* **do**
  for *speed, angle, coordinate in one trajectory* **do**
    calculate the average of absolute speed value spe
    calculate the sum of absolute angle value ang
    calculate the sum of direction value dir
  end
  $Spe_0[i] = spe$
  $Ang_0[i] = ang$
  $Dir_0[i] = dir$
end
for *each* spe_i, ang_i, dir_i *in* Spe_0, Ang_0, Dir_0 **do**
  calculate the absolute difference between spe_i and spe_{i_c} as spe_j, between ang_i and ang_{i_c} as ang_j, between dre_i and dre_{i_c} as dre_j,
  $Spe[i] = spe_j$
  $Ang[i] = ang_j$
  $Dir[i] = dir_j$
end
$Spe = Normalization(Spe)$
$Ang = Normalization(Ang)$
$Dir = Normalization(Dir)$
for *each* spe_k, ang_k, dir_k *in Spe, Ang, Dir* **do**
  $C[i] = w_1 * spe_k + w_2 * ang_k + w_3 * dir_k$
end

The angular distance $angDis(pr_i, Tr_j)$ reflects the degree of fluctuation in the inner direction of the two trajectories.The calculation method is:

$$angDis(pr_i, Tr_j) = |\sum_{k}^{N_{pr_i}} \theta_{k,k+1} - \sum_{k}^{N_{Tr_j}} \theta_{k,k+1}| \quad (14)$$

where $\theta_{k,k+1}$ indicates the angle between two adjacent points, N_{pr_i} represents the total number of angles between two points on a typical trajectory, N_{Tr_j} indicates the total number of angles between two points on the detected trajectory.

c) Direction Distance

The distance in the direction $dirDis(pr_i, Tr_j)$ indicates the difference in the overall deflection of the two trajectories in the direction of movement.The calculation method is:

$$dirDis(pr_i, Tr_j) = |\sum_{k}^{N_{pr_i}} d_{k,k+1} \cdot sin(\alpha) - \sum_{k}^{N_{Tr_j}} d_{k,k+1} \cdot sin(\beta)| \quad (15)$$

where $d_{k,k+1}$ indicates the distance between two adjacent points, N_{pr_i} represents the total number of distance between two points on a typical trajectory, the same, N_{Tr_j} indicates the total number of distance between two points on the detected trajectory, α and β respectively indicate the clockwise angle between the two points of the typical trajectory and the detected trajectory.

This calculation method combines the operating cost of the trajectory on multiple features, when the $WMFC(pr_i, Tr_j)$ smaller, the lower the operating cost of the trajectory, and vice versa. Therefore, the abnormality of the trajectory can be expressed as:

$$Abnormal(c) = WMFC(pr_i, Tr_j) \quad (16)$$

4) Evidence Theory Fusion Using the combination rule is to synthesize the basic credibility distribution from multiple sources to obtain a new credibility distribution as the output, which is used as the standard for the combination rule of multi-source information. In this article, we assume that a set of *N* trajectories between a pair of origin-destination nodes $< r_s, r_e >= \{Tr_1, Tr_2, Tr_3, ..., Tr_N\}$ is the recognition framework. These trajectories are a set of mutually exclusive and exhaustive possibilities. Our goal is to determine whether the suspicious trajectory has driving outlier based on the obtained trajectory confidence. For each trajectory, three abnormality indicators are calculated in this article and the pseudo-code of fusion anomaly extraction for each trajectory is shown in Algorithm 3.

$$Tr_i = \{A(t), A(d), A(c)\} \quad (17)$$

where $A(t), A(d), A(c)$ respectively represent the required abnormality $Abnormal(t), Abnormal(d), Abnormal(c)$.

Algorithm 3: Fusion Evidence Extraction

Input: Duration Evidence T, Distance Evidence D, Drivicng Cost Evidence C;
Output: Fusion Evidence bel;
initialize $K = \varnothing$, $bel = \varnothing$
for *each* t_i, d_i, c_i *in T, D, C* **do**
  calculate constant part of combination formula
  calculate $\widetilde{T}[i]$, $\widetilde{D}[i]$, $\widetilde{C}[i]$
  calculate the multiply value k_i
  $K[i] = k_i$
end
for *each* k_i *in K* **do**
  caculate fusion evidence value b_i
  $K[i] = b_i$
end

Since there is a lot of contradictory information that does not allow reasonable judgment, in order to obtain useful information between contradictory information, this article uses a combination formula:

$$\begin{cases} \widetilde{A(t)} = A(t) + \frac{1}{n(n-1)} \cdot e^{-\frac{2}{n(n-1)}} \cdot \phi \\ \widetilde{A(d)} = A(d) + \frac{1}{n(n-1)} \cdot e^{-\frac{2}{n(n-1)}} \cdot \phi \\ \widetilde{A(c)} = A(c) + \frac{1}{n(n-1)} \cdot e^{-\frac{2}{n(n-1)}} \cdot \phi \end{cases} \quad (18)$$

$$\phi = \sqrt[n]{A(t) \cdot A(d) \cdot A(c)} + \frac{A(t) + A(d) + A(c)}{n} \quad (19)$$

where *n* is the number of evidence sources. Therefore, the

combined confidence in the trajectory can be expressed as:

$$bel(Tr_i) = \widetilde{A(t)} \oplus \widetilde{A(d)} \oplus \widetilde{A(c)} = \frac{1}{K} \cdot \sum_{\Omega} [\widetilde{A(t)} \cdot \widetilde{A(d)} \cdot \widetilde{A(c)}] \tag{20}$$

$$\Omega = \widetilde{A(t)} \cap \widetilde{A(d)} \cap \widetilde{A(c)} = \{Tr_i\} \tag{21}$$

where K is the normalization constant.

$$K = \sum_{i=1}^{N} \widetilde{A(t)} \cdot \widetilde{A(d)} \cdot \widetilde{A(c)} \tag{22}$$

Abnormal Environment Perception By calculating the density of each segment, we can find the change of density value under each timestamp and achieve environmental awareness. Moreover, we can identify abnormal segments by detecting a sharp increase or decrease in density.

1) Extracting Edge Weight Sketch For each trajectory mapped in the road network, we can consider the intersection it passed as a set of evolving origin-destination subgraphs. We can combine the actual direction and the potential directions in the origin-destination trajectory and construct a directed weighted subgraph on this basis. Using the timestamp, we can combine them into evolving graphs. The effect of transforming the road network trajectories is shown in Fig. 8. The red arrow represents the current route, while the blue arrow represents the potential route.

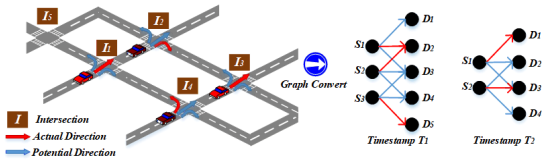


Fig. 8: The process of converting actual road network intersections and trajectories into evolving graph.

2) Calculating Density Given a graph $\mathcal{G}_t$, which represents the directed sub-graph and includes origin set $\mathcal{S}_t$ and destination set $\mathcal{D}_t$. It's density is defined in the way of $\rho(\mathcal{G}_t(\mathcal{S}_t, \mathcal{D}_t))$,

$$\rho(\mathcal{G}_t(\mathcal{S}_t, \mathcal{D}_t)) = \frac{\sum_{s_t \in \mathcal{S}_t, d_t \in \mathcal{D}_t} W_t}{|\mathcal{S}_t||\mathcal{D}_t|} \tag{23}$$

$$W_{t_1} = w_{r_1} d_{s_1,d_1} + w_{r_2} d_{s_1,d_2} + w_{r_3} d_{s_1,d_3} \tag{24}$$

where the w_{r_1}, w_{r_2} and w_{r_3} represent the weight of three directions, d_{s_i,d_j} represents the distance between the two intersections, thence the higher the total weight of edges, the greater its density. The specific algorithm can refer to Algorithm 4.

C. Classification

In this part, using the timestamp of each trajectory, we combine the evidence information of the outlier trajectory with the current geographic information. We assign the detected outlier trajectories, divided by timestamp, to each evolving graph. Based on the perception of environmental changes, we can classify the identified outlier trajectories as regular or irregular.

Algorithm 4: Calculating Density

Input: Origin Set $\mathcal{S}_t$, Destination Set $\mathcal{D}_t$;
Output: Sub-graph Density $\hat{\rho}$,A Stream of Anomaly Scores M;

initialize $\hat{\rho} = \varnothing$, $\mathcal{S}_t = \varnothing$, $\mathcal{D}_t = \varnothing$, $M = \varnothing$
for $\mathcal{G}_t \in G$ **do**
  for *each* s_i, d_i *in* $\mathcal{S}_t$, $\mathcal{D}_t$ **do**
    $s_i \leftarrow S_t, d_i \leftarrow D_t$
    calculate the distance between s_i and d_i
    calculate the weight of each part
    calculate the density value ρ_i
    $\hat{\rho}$[i] = ρ_i
    $s_{i+1} \leftarrow d_i$
    if $d_t = 0$ **then**
      break
    else
      calculate the anomaly score M_i
    end
  end
end
$M[i] = M_i$

V. EXPERIMENTS

In this section, we first introduce the dataset and experimental settings used in this paper. Then, we conduct extensive experiments to evaluate the effectiveness of the proposed framework.

A. Data Description

The multi-source city data includes road network data, taxi trajectory data, and POI data which aggregates human mobility features according to geographic similarity [22]. The file size after decompressing the trajectory dataset is approximately 21.4G.

B. Experiment Settings

To evaluate the effectiveness of our method and ensure that there are enough trajectories, we first set the functional area identified in the map to a radius of about 1000 meters and plan it as POI, so that the starting points of nearby taxi trajectories can be normalized to the same POI. Therefore, we use three of the following indicators *Accuracy*, *Recall* and *F1-Score* to evaluate the method.

With the process of identifying typical trajectories, we based on time and distance from the typical trajectory distribution, and examine the difference between the normal trajectory and the outlier trajectory, we can obtain the most typical trajectory by selecting the maximum difference Bel between the normal and outlier trajectories. We choose five typical trajectories as model benchmarks for comparison and use the three evaluation indicators accuracy, recall and F1-score to express the best results of a typical trajectory, the effect of each typical trajectory is shown in Fig. 9. While the difference reaches maximum, which indicates the effect of identifying anomalies in typical trajectories is better and the performance in different evaluation indicators reaches the best.

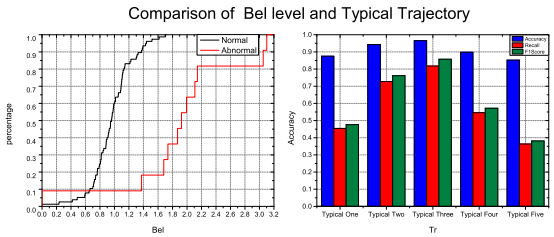


Fig. 9: Comparison difference between each trajectory and typical trajectory with the Bel of normal and abnormal.

As shown in Fig. 10, the x, y, and z coordinates respectively represent the values of w_1, w_2, and F1-Score. Since the weights of the three factors add up to one, the value of w_3 can be calculated from w_1 and w_2 (for example, when both w_1 and w_2 take the value 0, w_3 takes the value 1). For F1-Score, the value is expressed by z-axis height and color. The value more higher, the corresponding height more higher and the red color more darker. Otherwise, the value more lower, the corresponding height more lower and the blue color more darker. It can be noted that when w_1 and w_2 are both 1/3, the value of F1-Score is relatively maximum. In Algorithm 2, before the weighted fusion of the three factors, we normalize the three factors. Therefore, when the three factors take the same weight, each kind of evidence can be fully utilized.

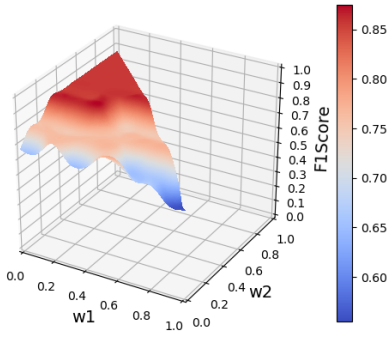
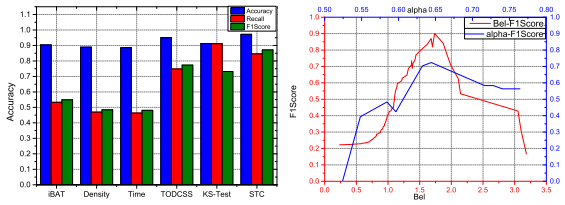


Fig. 10: Changes in F1-Score under different weights. While $w_1 = w_2 = w_3 = \frac{1}{3}$ F1-Score gets the optimal value.

To verify the effectiveness of the framework, we conducted comparative experiments with iBAT , density-based algorithm, time-dependent algorithm, TODCSS and KS-Test. Three sets of average results from Xi'an railway station to Xi'an Jiaotong University (Xingqing Campus), Xi'an Jiaotong University (Xingqing Campus) to Xi'an railway station and Xi'an coach station to Xi'an University of Architecture and Technology(Yanta Campus) are summarized in Tab. I, showing that the accuracy and F1-score obtained by using STC are higher. In terms of accuracy, our method has an average improvement of 6.40% compared to other methods. In terms of recall, although our method is slightly lower than the effect of KS-test, in general it has increased by 22.03%. In the most important assessment, F1-score, our method is superior to other methods and has improved by almost 26.76%.

TABLE I: The performance of each algorithm under the three evaluation methods.

Algorithms	Measurement		
	Accuracy	*Recall*	*F1-Score*
iBAT	*90.41%*	*53.33%*	*54.90%*
Density	*88.92%*	*46.97%*	*48.41%*
Time-Dependent	*88.51%*	*46.36%*	*48.10%*
TODCSS	*94.96%*	*74.85%*	*77.30%*
KS-Test	*91.27%*	***91.09%***	*73.17%*
STC	**97.21%**	*84.55%*	***87.14%***



(a) Comparison of methods (b) Comparison of parameters

Fig. 11: Comparison of the intuitive effects of various methods under three evaluation indicators and different parameter selection.

iBAT [7] Isolation-Based Anomalous Trajectory detection method, which proposed to put the problem of detecting abnormal driving trajectories into an easily solvable form.

Density [10] Compute the density-based outliers by finding all k neighbors in all possible trajectories.

Time-Dependent [13] Depending on the time, the most typical path of top-k can be used as a reference path.

TODCSS [16] Detect the trajectory outliers based on the distance calculation of the common slice subsequence.

KS-Test [21] Perform Kolmogorov-Smirnov Test statistics on the samples from the evaluation trajectory and the typical trajectory.

In our framework, we choose the real abnormal trajectory as the outlier trajectory to complete the accuracy, recall and F1-score measure. The result is shown in Fig. 11. Along the way, we decided to compare the F1-score with KS-test under different parameters, meanwhile, we can find that the results of our parameter selection are better than KS-test in different intervals. In the case of changing alpha and bel, the tuning effects of the two methods are compared. In a certain interval, our method achieves better results than KS-test.

C. Discussion

In this paper, we chose Xi'an Railway Station as the source node and Xi'an Jiaotong University (Xingqing Campus) as the destination node, the second set of experiments is from Xi'an coach station to Xi'an University of Architecture and Technology (Yanta Campus), the third set of experiments is from Xi'an Jiaotong University (Xingqing Campus) to Xi'an Railway Station (due to the limited space, so more results are omitted). We return the top ten outlier activities for each pair of source and destination nodes.

In Fig. 12 we have plotted all the trajectories. We highlight the outlier trajectories identified in these 10 trajectories in red

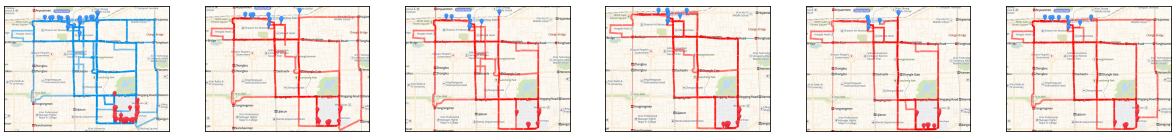
to make them more visible. For comparison, we also show the top 10 driving outlier trajectories in Fig. using a method based on the above methods. In Fig. 12, our combined method can identify partial diversion outliers more accurately than other methods. (a) All trajectories. (b) Top-10 Outlier Activies based on STC method. (c) Top-10 Outlier Activies based on iBAT. (d) Top-10 Outlier Activies based on Density. (e) Top-10 Outlier Activies based on Time. (f) Top-10 Outlier Activies based on TODCSS. The order of the method shown in Fig. 13. and Fig. 14. is the same as Fig. 12.

Algorithms proposed by predecessors detect anomalous trajectories well by analyzing the distance and density distribution between trajectories, while identifying multiple types of anomalies through in-depth analysis of urban traffic applications, including flow, section flow, trajectory, and its sub-trajectory, which greatly improves the effectiveness of anomaly detection. However, due to the time-varying sparse distribution of trajectory extraction and matching, it is still impossible to judge whether there is an objective reason for abnormal trajectory detection because the methods do not describe the maneuverability of driving behavior, which is constrained by the actual road network. Meanwhile, the above algorithms all analyze the problem by individuals and ignore that taxis are an inseparable group that affect each other, the relationship between upstream and downstream vehicles is ignored.

For one of the trajectories identified by our method, as shown in Fig. 15, our method recognizes it as a outlier trajectory, while other methods consider it as a normal trajectory. The trajectory is described in detail as follows: first head south, then turn northeast and go around Xi'an East Second Ring Road to reach the destination. In reality, the starting section and the ending section of the trajectory are part of the common trajectory. Both iBAT and Density methods divide the trajectory area into grids and compare them in grids. The iBAT method classifies trajectories by randomly selecting trajectories and comparing overlapping grids, so it is not stable. The density method classifies trajectories by comparing the number of trajectories in the grid, but the trajectory is affected by the first and last road segments, which causes it to classify the trajectory as normal. The time dependent method classifies the trajectory by comparing the total time consumption of the trajectory. The vehicle speed of the trajectory is higher, but the time is not the highest, so it recognizes the trajectory as normal. The TODCSS method classifies trajectories by comparing the length of the trajectory. The length of the trajectory is higher but not the longest trajectory, so the trajectory is recognized as normal. For our method, the comprehensive abnormality degree is higher than other trajectories because the vehicle makes a U-turn and a detour so our method considers it as an abnormal trajectory.

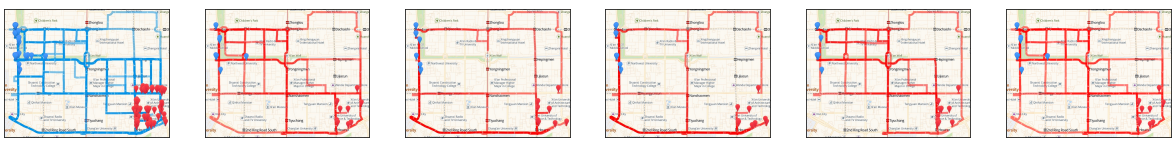
In the course of analyzing outlier trajectories, we examined the objective geographic factors of such phenomena in more detail. As taxis concentrate near the station to pick up passengers, the density in the initial sub-frame is relatively high, but as taxis travel to the destination, the density of the sub-frame roughly shows a trend of dissipation. However, in t_{12} there was a sudden increase in density. By gradually linking the global context with the local one [23], we construct a coarse-to-fine hierarchical trajectory framework. Since the sharp increase in density represents more taxi trajectories, we can assume that there is congestion in this segment, so the driver chooses other trajectories that reduce time at the expense of distance. We judge this outlier trajectory to be caused by objective factors rather than subjective reasons. The actual situation is shown in Fig. 16. In other abnormal trajectory detection methods, they mainly focus on the total time, total distance, overlap and other data of the trajectory for classification. For example, the classification performance of the method Time-Dependent and the method TODCSS is affected by the total time and the total distance of the trajectory, respectively. However, the abnormality of the data may be influenced by the condition of traffic congestion, rather than the subjective intention of the taxi driver. Therefore, this method uses density changes to obtain the actual traffic congestion condition, realizes the environmental perception during the taxi ride, and further classifies the abnormal trajectories obtained by the STC method. Other abnormal trajectory detection methods only process the value of the trajectory data and ignore the implicit collaboration information in the data, and therefore cannot accurately hide the taxi drivers' subjective diversion behavior.

However, the performance of this method strongly depends on the selected typical trajectory. If the dataset is small or the



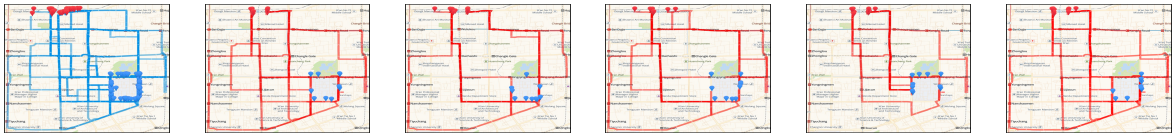
(a) (b) (c) (d) (e) (f)

Fig. 12: Fraud driving activities comparison from Xi'an Railway Station to Xi'an Jiaotong University.



(a) (b) (c) (d) (e) (f)

Fig. 13: Fraud driving activities comparison from Xi'an coach station to Xi'an University of Architecture and Technology.



(a) (b) (c) (d) (e) (f)

Fig. 14: Fraud driving activities comparison from Xi'an Jiaotong University to Xi'an Railway Station.

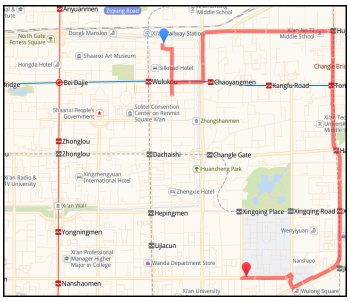


Fig. 15: The trajectory which is correctly identified as an anomaly while the existing method is incorrectly classified.

range of the start and end points POI nodes is too large, the wrong typical trajectory may be found, and the performance of the method will decrease rapidly. In addition, the algorithm has problems such as low efficiency and large amount of work for manually marking samples when processing a large number of trajectories. Considering the above limitations, we will use the algorithm in the cloud in future work. We combine the state-of-the-art algorithms to improve the accuracy of screening abnormal trajectories in small datasets. Meanwhile, we will reduce the performance problems caused by POI changes to improve the anti-interference ability of the model.

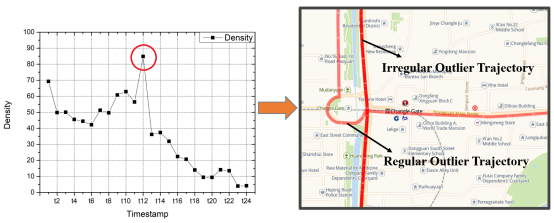


Fig. 16: Different outlier trajectory performance wtih high-density conditions under the classification in C-IoVs.

VI. CONCLUSION

Outlier detection algorithms have long been widely used for urban traffic data. Based on multiple urban data sources, such as taxi trajectory data under real-time V2V/V2R communication and urban road network information, we propose an algorithm for detecting irregular outlier trajectories in combination with temporal-spatial data considering C-IoVs environment. The algorithm obtains a pair of typical trajectories of origin-destination nodes from the taxi trajectories of C-IoVs and the cooperation of participants as a benchmark. Then, the three aspects of time, distance and travel cost are combined to obtain the outlier trajectory. By combining the objective factors of regional anomalies, we can explore the motivation for the abnormal trajectory more deeply. In future work, we will further improve the applicability and anti-interference.

REFERENCES

[1] W. Wang, J. Chen, J. Wang, J. Chen, and Z. Gong, "Geography-aware inductive matrix completion for personalized point-of-interest recommendation in smart cities," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4361–4370, 2019.

[2] X. Han, G. Shen, X. Yang, and X. Kong, "Congestion recognition for hybrid urban road systems via digraph convolutional network," *Transportation Research Part C: Emerging Technologies*, vol. 121, p. 102877, 2020.

[3] X. Zhou, X. Xu, W. Liang, Z. Zeng, S. Shimizu, L. T. Yang, and Q. Jin, "Intelligent small object detection based on digital twinning for smart manufacturing in industrial cps," *IEEE Transactions on Industrial Informatics*, 2021.

[4] M. A. Javed and E. B. Hamida, "On the interrelation of security, QoS, and safety in cooperative ITS," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 7, pp. 1943–1957, 2016.

[5] C. Chen, D. Zhang, P. S. Castro, N. Li, L. Sun, and S. Li, "Real-time detection of anomalous taxi trajectories from GPS traces," in *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Springer, 2011, pp. 63–74.

[6] W. Wang, F. Xia, H. Nie, Z. Chen, Z. Gong, X. Kong, and W. Wei, "Vehicle trajectory clustering based on dynamic representation learning of internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[7] D. Zhang, N. Li, Z.-H. Zhou, C. Chen, L. Sun, and S. Li, "iBAT: Detecting anomalous taxi trajectories from gps traces," in *Proceedings of the 13th International Conference on Ubiquitous Computing*, 2011, pp. 99–108.

[8] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2020.

[9] M. Gupta, J. Gao, Y. Sun, and J. Han, "Integrating community matching and outlier detection for mining evolutionary community outliers," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012, pp. 859–867.

[10] Y. Ge, H. Xiong, C. Liu, and Z.-H. Zhou, "A taxi driving fraud detection system," in *2011 IEEE 11th International Conference on Data Mining*. IEEE, 2011, pp. 181–190.

[11] J. Wang, Y. Yuan, T. Ni, Y. Ma, M. Liu, G. Xu, and W. Shen, "Anomalous trajectory detection and classification based on difference and intersection set distance," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 2487–2500, 2020.

[12] M. Xu, J. Wu, H. Wang, and M. Cao, "Anomaly detection in road networks using sliding-window tensor factorization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4704–4713, 2019.

[13] J. Zhu, W. Jiang, A. Liu, G. Liu, and L. Zhao, "Effective and efficient trajectory outlier detection based on time-dependent popular route," *World Wide Web*, vol. 20, no. 1, pp. 111–134, 2017.

[14] X. Zhou, Y. Ding, F. Peng, Q. Luo, and L. M. Ni, "Detecting unmetered taxi rides from trajectory data," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 530–535.

[15] C. Chen, D. Zhang, P. S. Castro, N. Li, L. Sun, S. Li, and Z. Wang, "iBOAT: Isolation-based online anomalous trajectory detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 806–818, 2013.

[16] Q. Yu, Y. Luo, C. Chen, and X. Wang, "Trajectory outlier detection approach based on common slices sub-sequence," *Applied Intelligence*, vol. 48, no. 9, pp. 2661–2680, 2018.

[17] G. Xue, M. Zhong, J. Li, J. Chen, C. Zhai, and R. Kong, "Dynamic network embedding survey," *arXiv preprint arXiv:2103.15447*, 2021.

[18] X. Kong, S. Tong, H. Gao, G. Shen, K. Wang, M. Collotta, I. You, and S. Das, "Mobile edge cooperation optimization for wearable internet of things: a network representation-based framework," *IEEE Transactions on Industrial Informatics*, 2020.

[19] D. Eswaran, C. Faloutsos, S. Guha, and N. Mishra, "Spotlight: Detecting anomalies in streaming graphs," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1378–1386.

[20] Y. Li, J. Zhou, J. Tian, X. Zheng, and Y. Y. Tang, "Weighted error entropy-based information theoretic learning for robust subspace representation," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[21] T. He, J. Bao, R. Li, S. Ruan, Y. Li, C. Tian, and Y. Zheng, "Detecting vehicle illegal parking events using sharing bikes' trajectories." in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 340–349.

[22] G. Shen, Z. Zhao, and X. Kong, "GCN2CDD: A commercial district discovery framework via embedding space clustering on graph convolution networks," *IEEE Transactions on Industrial Informatics*, 2020.

[23] Y. Li, R. Liang, W. Wei, W. Wang, J. Zhou, and X. Li, "Temporal pyramid network with spatial-temporal attention for pedestrian trajectory prediction," *IEEE Transactions on Network Science and Engineering*, 2021.